LITERATURE REVIEW: ADVANTAGES AND DISADVANTAGES OF BLACK BOX AND WHITE BOX TESTING METHODS

Asri Maspupah[1*]

Department of Computer Engineering and Informatics[1]
Politeknik Negeri Bandung, Bandung, Indonesia[1]
polban.ac.id[1]
asri.maspupah@polban.ac.id[1*]

(*) Corresponding Author

Abstract *— Software testing methods play a crucial role in ensuring the quality, security, and performance of applications. The two main approaches often used are black box and white box testing. This article presents a comprehensive literature review of the characteristics, advantages and disadvantages of both testing methods with the aim of providing in-depth insights for practitioners and researchers in the field of software engineering. The research method used is Systematic Literature Review (SLR), which is a technique of collecting information through the process of searching for literature based on research questions, filtering search results, evaluating, and analyzing relevant research from various academic database sources to gain a comprehensive understanding. Through the analysis of various literature studies, the results of this research are the characteristics of each testing method that is more effectively used in best practice to achieve efficient testing with the coverage of literature review sources in the range of 2020 to 2024. The review of these characteristics is based on seven criteria, namely: test focus, test base, tester's point of view, test case design, program error identification method, program code knowledge, and test effectiveness. In addition, this article presents a comparison of the two methods so that the differences in testing methods can be clearly recognised in terms of how they are implemented. Based on the results of review literature, both methods are better used together in order to ensure that the functional and internal code structure of the software is functioning properly.*

Keywords*: black box testing, software testing, systematic literature review, white box testing.*

Intisari—*Metode pengujian perangkat lunak memainkan peran krusial dalam memastikan kualitas, keamanan, dan kinerja aplikasi. Dua pendekatan utama yang sering digunakan adalah pengujian black box dan white box. Artikel ini menyajikan tinjauan literatur yang komprehensif mengenai karakteristik, kelebihan dan kekurangan kedua metode pengujian dengan tujuan memberikan wawasan mendalam bagi praktisi dan peneliti di bidang rekayasa perangkat lunak. Metode penelitian yang digunakan adalah Systematic Literature Review (SLR), yakni teknik pengumpulan informasi melalui proses pencarian literatur berdasarkan research question, filtering hasil pencarian, evaluasi, dan analisis penelitian yang relevan dari berbagai sumber basis data akademik untuk mendapatkan pemahaman komprehensif. Melalui analisis berbagai studi literatur, hasil penelitian ini berupa karakteristik masing-masing metode pengujian yang lebih efektif digunakan secara best practice guna mencapai pengujian efisien dengan cakupan sumber literatur review berada pada rentang tahun 2020 sampai tahun 2024. Tinjauan karakteristik tersebut berdasarkan tujuh buah kriteria, yaitu: fokus pengujian, basis pengujian, sudut pandang penguji, desain kasus uji, cara identifikasi kesalahan program, pengetahuan kode program, dan efektivitas pengujian. Selain itu, artikel ini menyajikan perbandingan kedua metode sehingga perbedaan metode pengujian dapat diketahui dengan jelas dari sisi cara penerapannya. Berdasarkan hasil literatur review, kedua metoda lebih baik digunakan secara bersamaan agar dapat memastikan fungsional dan struktur kode internal perangkat lunak telah berfungsi dengan baik.*

Kata Kunci*: pengujian kotak hitam, pengujian perangkat lunak, tinjauan literatur sistematis, pengujian kotak putih.*

INTRODUCTION

Software testing is an essential part of the software development life cycle that must be conducted to ensure quality and compliance with user requirements(Kusum et al., 2024). The purpose of testing is to identify errors, validate functionality, and evaluate the performance of software before it is handed over to the end users (Taley & Pathak, 2020). An effective testing strategy includes considerations of objectives, timing, resources, and testing environments (Arifandi et al., 2022). Although testing does not guarantee complete quality, it can enhance user confidence in the software (Dhaifullah et al., 2022). Beginner developers often overlook thorough testing, whereas it is important to improvise and choose methods that align with the needs and complexity of the system.

There are several testing methods, including black-box, white-box, and grey-box testing (Dhaifullah et al., 2022). Among various software testing techniques, black box testing and white box testing are the two primary approaches that are frequently used. Black box testing emphasizes assessing a system's functionality according to its requirements without analyzing the internal code structure (Sholeh et al., 2021). Meanwhile, white box testing emphasizes examining program code by analyzing its internal logic and code paths (Golian et al., 2022). Each of these methods has its own characteristics, advantages, and disadvantages. Black box testing is more efficient for a smaller number of test cases, while white box testing is better suited for higher testing frequencies (Wintana et al., 2022).

Black box testing is particularly suitable for beginner testers who may lack programming skills (Uminingsih et al., 2022). This method is highly effective for evaluating software behavior to identify errors in functions, interfaces, data models, and external data access(Shadiq et al., 2021). Equivalence class partitioning is the most commonly used technique in black box testing; it involves grouping input data into positive and negative test categories to minimize the number of test cases (Hamzah & Voutama, 2023). Although black box testing can uncover system deficiencies, it may not capture all errors, especially those related to the internal code structure.

White box testing is employed to identify anomalies, verify internal operations, and ensure that expected results are achieved (Kalfin et al., 2024). It is particularly effective in unit testing, with error detection rates reaching 65% of defects in the software being tested (Mishra et al., 2020). Basis path testing is a widely used technique within white box testing. This method is implemented to guarantee comprehensive coverage of the control structure (Sasmito, 2020; Kalfin et al., 2024;). It involves creating flowcharts and flowgraphs, as well as calculating cyclomatic complexity to determine testing scenarios (Sasmito, 2020).

When comparing white box testing to black box testing, white box testing is generally more effective for routine tests that require repetition. In contrast, black box testing is better suited for tests that are infrequently repeated. Each method has its own advantages and disadvantages, which depend on specific needs. The choice between black box testing and white box testing should be guided by the particular requirements of the project, as each method possesses distinct characteristics, strengths, and limitations (Wintana et al., 2022).

As technology evolves, software development becomes increasingly complex, requiring the integration of various platforms and the management of vast volumes of data. Concurrently, the rising threat of cybersecurity has made software security testing a top priority. Consequently, effective and efficient software testing is essential to ensure software quality, reliability, and security (Dhaifullah et al., 2022). Furthermore, trends in software development methodologies, such as Agile and DevOps, have transformed the approach to software development and testing. DevOps, which emerged in 2009, combines Agile Systems and Agile Operations, fostering collaboration between development and infrastructure teams throughout the software lifecycle (Riyadi, 2023).

Proper testing is crucial to ensure software quality and minimize operational issues following deployment. Therefore, both methods are relevant to discuss in the context of a literature review. The primary focus of this study is a literature review of black box and white box testing approaches, with the aim of comparing the two methods to understand their characteristics, advantages, and limitations. The research method employed is a systematic literature review (SLR), which examines the application of black box and white box testing in software development. The SLR consists of three main processes: planning, conducting, and reporting, to answer the research question, "How do the effectiveness of black box testing and white box testing methods compare in software development?" The research findings are expected to assist software developers in selecting the appropriate testing method based on factors such as system complexity, testing basis, testing objectives, time constraints, and the type of software development.

Preliminary findings from the literature indicate that while black box testing is more effective for functional testing, white box testing

excels in identifying internal logic errors, making both methods complementary depending on the project's complexity and stage. Several studies on black-box and white-box testing have been conducted, including those by Dhaifullah (2022), Arifandi (2022), Wintana (2022), and Golian (2022).

In his research, Dhaifullah examined testing techniques for black-box, white-box, and grey-box testing (Dhaifullah, 2022). Arifandi discussed the techniques, advantages, and benefits of black-box, white-box, and grey-box testing (Arifandi, 2020). Additionally, Arifandi conducted a comparison of testing techniques in white-box and grey-box methods, analyzing fundamental aspects such as internal knowledge, test execution time, suitability for algorithm evaluation, and granularity level. Meanwhile, Wintana investigated the effectiveness of white-box and black-box testing in unit testing through a case study of decode and encode functions in a time unit (Wintana, 2022). Lastly, Golian explored various black-box and white-box testing approaches applicable to web application testing.

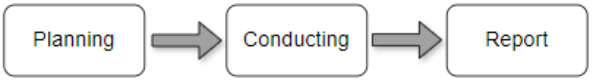
The distinction between this study and prior research is highlighted in the comprehensive literature review that explores the differences between black-box and white-box testing, considering their characteristics, advantages, and limitations. From this review, comparison criteria were established, including definitions, objectives, test basis, perspectives, techniques, suitability for specific applications, strengths, weaknesses, and test executors. The novelty of this research lies in the more in-depth comparison of the two testing methods.

MATERIALS AND METHODS

In this article, the research method used is the Systematic Literature Review (SLR). SLR is a methodological approach to synthesizing relevant empirical evidence from various sources on a specific topic (Reyna et al., 2020). The SLR process involves a structured approach to searching, assessing, and analyzing relevant studies (Amin et al., 2022; Mengist et al., 2020). SLR can be applied in various fields, including environmental science, mathematics education, and cyber risk assessment (Amin et al., 2022; Iskandar & Juandi, 2022; Mengist et al., 2020). This method allows researchers to systematically evaluate existing knowledge, identify trends, and discover gaps in the literature (Mengist et al., 2020).

The SLR method was selected due to its comprehensive review process, which relies on empirical evidence previously gathered by researchers. This approach minimizes bias in the selection of studies and data analysis (Maulana et al., 2024). Furthermore, the SLR process is structured, beginning with the formulation of research questions and culminating in data synthesis (Wahyudin, 2020). These characteristics enhance the reliability and relevance of the research findings in relation to the research questions. The SLR method enables researchers to systematically analyze pertinent literature, allowing them to identify knowledge gaps and develop new, innovative research questions (Wazzan et al., 2021).

SLR generally follows a three-stage process: Planning, Conducting, and Reporting (Amin et al., 2022, 2023). The flow of SLR is illustrated in Figure 1. SLR can combine qualitative and quantitative analysis techniques and is conducted objectively and transparently, providing a comprehensive understanding of the research landscape (Iskandar & Juandi, 2022; Mengist et al., 2020).



Source : (Arifandi et al., 2022)

Figure 1. Process Flow SLR

A. *Planning*

The planning stage involves defining the research question (RQ) and developing a review protocol (Arifandi et al., 2022). The RQ is a part of the SLR review protocol that is used to guide the literature search and extraction process. The analysis and synthesis of data generated from the SLR provide answers to the previously established RQ. A good RQ is one that is beneficial, measurable, and aims to understand the latest research on a specific topic. The formulation of the RQ is based on five elements known as PICOC (Mengist et al., 2020). The explanation of PICOC is presented in Table 1.

Tabel 1. Elements of RQ Composition Using the PICOC Approach

Elemen	Description	Application in Research
Patient, Population or Problem (P)	Target data collection to be investigated	The black box and white box testing approaches comply with ISO standards.
Intervention (I)	Detailed Aspects of Investigation or Issues of Interest to the Researcher	Testing methods based on different approaches
Comparison of Intervention (C)	Investigative Aspects of the Intervention (I) to be Compared	Comparison of the characteristics of testing approaches according to ISO standards
Outcome (O)	Impact and results of the intervention (I)	Advantages and disadvantages of the testing approaches
Context (C)	Investigative environment Settings.	Academic field studies.

Source: (Research Results, 2024)

Based on Table 1, the RQs can be formulated as follows:

RQ-1: What are the characteristics of black box and white box testing approaches?

RQ-2: What are the advantages of the black box and white box approaches?

RQ-3: What are the disadvantages of the black box and white box approaches?

Developing an SLR protocol is a crucial step in conducting an SLR. The SLR protocol is a plan that outlines the procedures for systematic review (Fitriani & Kuswadi, 2021). The application of SLR has been implemented in various systems, including sustainable production in mountain watershed areas (Fitriani & Kuswadi, 2021), flood disaster detection using machine learning (Faiza. Intan Mayla et al., 2022), and other software engineering issues. An SLR protocol typically includes seven elements: background, research questions, search terms, selection criteria, quality assessment, data extraction, and synthesis strategies (Fitriani & Kuswadi, 2021). The explanation and application of the SLR protocol are shown in Table 2.

Table 2. SLR Protocol

Protocol	Description	Application in Research
background	The background of the research explains the context and importance of the researched topic.	The choice of the appropriate testing approach for a project depends on the suitability of the project conditions with the characteristics of each approach.
research questions	Research questions (RQs) are formulated to guide the literature search and extraction process.	RQ-1, RQ-2 an d RQ-3
search terms	The search terminology is determined based on the RQ, which consist of relevant keywords related to the study topic, including search strategies in academic databases.	• Characteristics of black box and white box testing. • Advantages of Black Box and White Box Testing. • Disadvantages of Black Box and White Box Testing.
selection criteria	The selection criteria consist of inclusion and exclusion criteria for choosing literature.	Inclusion and exclusion criteria are presented in Table 4.
Quality Assessment	A quality checklist and procedures are used to assess the validity and reliability of the selected studies.	Quality assessment criteria for the literature are shown in Table 5.
data extraction strategies	Strategies for data extraction involve collecting essential information from the selected literature.	Data extraction is based on the following categories: • characteristics of black box testing; • characteristics of white box testing; • advantages of black box testing; • advantages of white box testing; • disadvantages of black box testing; • disadvantages of white box testing
synthesis strategies	Data synthesis strategies involve analyzing and integrating findings from various studies.	Comparisons of approaches are based on the categories of extraction strategies.

Source: (Research Results, 2024)

B. Conducting

During the implementation stage, the literature search process (search string) is carried out based on keywords from various academic database sources (Iskandar & Juandi, 2022). The search process consists of several steps: selecting the digital database sources, defining keywords, executing the search process, and retrieving the search results (Arifandi et al., 2022). The determination of the digital database set aims to ensure that the search process is relevant to the related discussion topic. The database sources used include ResearchGate, IEEE Xplore, ScienceDirect, and Google Scholar, which index national and international scientific journals. Meanwhile, the keywords are created based on PICOC (Table 1) following the pattern in Table 3.

Table 3. Search Keywords

Languange	Keyword
English	(software) AND (testing* OR measuring OR examining) AND (technique* OR approach OR procedure OR characteristics OR advantages OR disadvantages) AND (black box OR white box)
Bahasa	(pengujian perangkat lunak) AND (pendekatan OR metode OR karakteristik OR kelebihan OR kekurangan) AND (black box OR white box)

Source: (Research Results, 2024)

A literature search was conducted by querying selected digital repositories with specific keywords, filtering the results based on the research question, and storing the retrieved data. Next, the literature screening process involves the use of inclusion and exclusion criteria, often utilizing tools such as EndNote and the PRISMA flow diagram (Amin et al., 2022, 2023). The inclusion and exclusion criteria for this study are detailed in Table 4. The application of these criteria is intended to evaluate the quality of the chosen studies to guarantee the validity and reliability of the search results.

Table 4. Inclusion and Exclusion Criteria

Inclusion	Exclusion
The literature discusses topics related to the characteristics, advantages, and disadvantages of black box and white box testing approaches.	The scope of the literature topics is too broad, exceeding the research study's scope.
The literature is written in Indonesian or English.	The literature is not written in Indonesian and English.
The literature was published between 2020 and 2024.	The literature was published before 2020.
The literature is published in scientific journals, conference proceedings, books, and research reports.	The literature is published without undergoing a peer review process, such as in library repositories or magazines.

Source: (Research Results, 2024)

After the literature screening, the quality assessment process of the search results is conducted to obtain literature relevant to the topic. The evaluation criteria for this study are shown in Table 5. Each piece of literature is assigned a weight: 1 (the literature meets all criteria), 0.5 (the literature partially meets the criteria), or 0 (the literature does not meet any of the criteria) (Maulana et al., 2024). If the literature meets all criteria, the highest total weight score is 6.

Table 5. Quality Assessment Criteria

Assessment	Criteria
QA1	Do the articles discuss the characteristics of the black box testing approach?
QA2	Do the articles discuss the characteristics of the black box testing approach?
QA3	Do the articles discuss the advantages of the black box testing approach?
QA4	Do the articles discuss the advantages of the white box testing approach?
QA5	Do the articles discuss the disadvantages of the black box testing approach?
QA6	Do the articles discuss the disadvantages of the white box testing approach?

Source: (Research Results, 2024)

The final step, data analysis and synthesis, aims to formulate the main findings and draw conclusions that can answer the research questions (Iskandar & Juandi, 2022; Suryani et al., 2021). The mechanism for conducting data analysis and synthesis follows the defined SLR protocol.

C. *Reporting*

The reporting stage summarizes findings that contribute to the development of the framework (Amin et al., 2023). Reporting can be done in the form of scientific papers or articles published in journals, presented at seminars, or structured as a literature review in a thesis or dissertation.

The reporting phase involves detailing the results obtained at each stage of the SLR, as illustrated in Table 4. This phase also includes the preparation of publicly accessible reports, such as journal articles. In an SLR, the creation of a journal article represents the final step, aimed at generating research output (Mengist, 2020). The general steps for report creation are outlined below.

1. This section provides a detailed description of the primary procedures, including an explanation of the search methods, an outline of the selection process, a description of data extraction, and an overview of data analysis.
2. Presentation of Results: This section includes a summary of the main findings, data presentation through tables or graphs, implications of the obtained findings, and an identification of limitations.

RESULTS AND DISCUSSION

A. *Literature Search Process*

The results of the literature search are categorized based on search criteria, as shown in Table 6. A total of 15 pieces of literature were successfully collected, with each piece of literature matched to the search categories.

Table 6. Literature Search Results

Search Criteria	Count	Source
Characteristics of Black Box Testing	9	(Tanli et al., 2020); (Mishra et al., 2020); (Salim et al., 2021); (Dhaifullah et al., 2022); (Golian et al., 2022); (Kurniawan et al., 2022); (Wintana et al., 2022); (Praniffa et al., 2023);
Characteristics of White Box Testing	15	(Tanli et al., 2020); (Mishra et al., 2020); (Salim et al., 2021); (Wijaya & Astuti, 2021); (Weber et al., 2021); (Dhaifullah et al., 2022); (Golian et al., 2022); (Hartono & Sugiarti, 2022); (Santi et al., 2022); (Sie et al., 2022); (Susilawati et al., 2022); (Uminingsih

Search Criteria	Count	Source
		et al., 2022); (Wintana et al., 2022); (Praniffa et al., 2023); (Hamzah & Voutama, 2023)
Advantages of Black Box Testing	10	(Mishra et al., 2020); (Salim et al., 2021); (Wijaya & Astuti, 2021); (Weber et al., 2021); (Dhaifullah et al., 2022); (Golian et al., 2022); (Sie et al., 2022); (Susilawati et al., 2022); (Uminingsih et al., 2022); (Wintana et al., 2022); (Praniffa et al., 2023); (Hamzah & Voutama, 2023)
Advantages of White Box Testing	7	(Salim et al., 2021); (Dhaifullah et al., 2022); (Golian et al., 2022); (Uminingsih et al., 2022); (Wintana et al., 2022); (Praniffa et al., 2023); (Hamzah & Voutama, 2023)
Advantages of White Box Testing	6	(Salim et al., 2021); (Weber et al., 2021); (Dhaifullah et al., 2022); (Golian et al., 2022); (Wintana et al., 2022); (Praniffa et al., 2023); (Hamzah & Voutama, 2023)
Disadvantages of Black Box Testing	4	(Salim et al., 2021); (Sie et al., 2022); (Wintana et al., 2022); (Praniffa et al., 2023);
Disadvantages of White Box Testing	4	(Salim et al., 2021); (Sie et al., 2022); (Wintana et al., 2022); (Hamzah & Voutama, 2023)

Source: (Research Results, 2024)

B. Literature Search Selection and Assessment Process

The selection process for the literature search, based on inclusion and exclusion criteria, resulted in the exclusion of 15 pieces of literature. During the literature search, the exclusion criteria were taken into account, resulting in the initial filtering of literature with broader coverage, works not written in Indonesian or English, and publications released before 2020. Subsequently, each piece of literature was assessed using quality assessment criteria to obtain literature that is truly relevant to the research topic. The results of the quality assessment of the literature are shown in Table 7.

The literature assessment scores presented in Table 7 reflect the completeness of the substance concerning the quality assessment criteria outlined in Table 5. Many of the reviewed studies discuss the characteristics of both testing methods; however, only a few address the strengths and weaknesses of each method. Notably, one study, Kurniawan's 2022 research, does not align with the research review and received a score of zero because it failed to address any of the criteria topics listed in Table 5. The literature assessment scores will significantly influence the thoroughness of the research review in elucidating the characteristics, strengths, and weaknesses of black box and white box testing methods.

Table 7. Results of the Quality Assessment of the Literature

Reference	Year	Article Title	QA1	QA2	QA3	QA4	QA5	QA6	Total
(Tanli et al., 2020)	2020	Black-box approach for software testing based on fat-property	1	0	0	0	0	0	1 (17%)
(Mishra et al., 2020)	2020	White Box Testing Using Genetic Algorithm—An Extensive Study	1	1	0	0	0	0	2 (33%)
(Salim et al., 2021)	2021	Comparative Analysis of Software Testing Techniques for Mobile Applications	1	1	1	1	1	1	6 (100%)
(Wijaya & Astuti, 2021)	2021	Pengujian Blackbox Sistem Informasi Penilaian Kinerja Karyawan PT Inka (Persero) Berbasis Equivalence Partitions	1	0	0	0	0	0	1 (17%)
(Weber et al., 2021)	2021	White-Box Performance-Influence Models_A Profiling and Learning Approach	1	1	1	0	0	0	1 (50%)
(Dhaifullah et al., 2022)	2022	Survei Teknik Pengujian Software	1	1	0	0	0	0	2 (33%)
(Golian et al., 2022)	2022	Black And White-Box Unit Testing For Webapplications	1	1	1	1	0	0	4 (67%)
(Hartono & Sugiarti, 2022)	2022	Perbandingan Metode Equivalence Partitions dan Boundary Value Analysis Pada Pengujian-Black Box	1	0	0	0	0	0	1 (17%)
(Kurniawan et al., 2022)	2022	Pengujian Sistem Informasi Manajemen Siswa Berbasis Website Menggunakan Metode Black Box Dan White Box	0	0	0	0	0	0	0 (0%)
(Uminingsih et al., 2022)	2022	Pengujian Fungsional Perangkat Lunak Sistem Informasi Perpustakaan Dengan Metode Black Box Testing Bagi Pemula	1	1	1	0	0	0	3 (50%)

Reference	Year	Article Title	QA1	QA2	QA3	QA4	QA5	QA6	Total
(Santi et al., 2022)	2022	Black Box Testing with Equivalence Partitioning and Boundary Value Analysis Methods (Study Case: Academic Information System of Mataram University)	1	0	0	0	0	0	1 (17%)
(Sie et al., 2022)	2022	Pengujian White Box Testing Terhadap Website Room Menggunakan Teknik Basis Path	1	1	0	0	0	0	2 (33%)
(Susilawati et al., 2022)	2022	Pengujian Black Box Aplikasi Penjualan Pupuk Bersubsidi Menggunakan Teknik Equivalence Partitioning	1	0	0	0	0	0	1 (17%)
(Wintana et al., 2022)	2022	Analisis Perbandingan Efektifitas White-Box Testing dan Black-Box Testing	1	1	1	1	1	1	6 (100%)
(Praniffa et al., 2023)	2023	Pengujian Black Box Dan White Box Sistem Informasi Parkir Berbasis Web	1	1	1	1	1	0	5 (83%)
(Hamzah & Voutama, 2023)	2023	A Black Box Tesing for a University Database Application Using Equivalence Partition Method	1	1	1	1	0	1	5 (83%)

Source: (Research Results, 2024)

C. *Data Analysis*

RQ-1: What are the characteristics of black box and white box testing approaches?

Black box testing is a testing method that evaluates the functionality of software without considering the internal structure or code implementation. It is often referred to as functional testing. In this method, testers only interact with the user interface and verify the output produced based on specific inputs according to the software requirements specification. The characteristics of black box testing are shown in Table 8.

Table 8. Characteristics of Black Box Testing

Criteria	Characteristics
Focus of Testing	• Testing the input and output of the system is essential to ensuring that the software functions according to the specifications. • Does not concern itself with how that functionality is implemented.
Basis of Testing	• A specification-based testing approach. • Testing is based on the software requirements specifications. • Testers use software documentation, such as requirement specifications, functional specifications, and system design, to formulate test scenarios.
End-User Perspective Approach	• Testing the software from the end user's perspective. • Simulating real usage by the end user is essential to ensuring that the system functions correctly.
Test Case Design	• Test cases are created based on defined usage scenarios, test data, and input/output conditions. • Common techniques used include equivalence partitioning, boundary value analysis, and decision table testing.
Identification of Program Errors	• The focus of testing is to discover defects in the functional behavior of the software. • It helps identify defects in interface errors, functional behavior, and input/output processes.
Knowledge of Program Code	• Testers do not need to know the source code of the application being tested. • Testing is conducted independently from code development.
Effectiveness of Testing	Extremely effective in testing the functional aspects of software to ensure they are working as expected.

Source: (Research Results, 2024)

White box testing is a testing method that validates the internal structure and logic of software. White box testing is also known as structural testing or glass box testing. Testers have in-depth knowledge of the code, algorithms, and architecture of the system being tested. This testing approach allows testers to verify the program's control flow, data flow, boundary conditions, and execution paths within the code. The characteristics of white box testing are shown in Table 9.

Table 9. Characteristics of White Box Testing

Criteria	Characteristics
Focus of Testing	Testing is aimed at validating the structure of the application's program code, including logical flows, conditions, loops, and execution paths.
Basis of Testing	• A code-based testing approach. • Testers have full access to the program code. • Testing involves thorough code analysis, including testing logical conditions, loops, and recursion.

Criteria	Characteristics
End-User Perspective Approach	• An approach based on the developer's perspective. • Testing includes validating conditions (e.g., if, switch) and loops (e.g., for, while) to ensure all branches and iterations work correctly. • Testing involves tracing data from input, data processing, and output processes to ensure that the data has been processed correctly. • Testing covers all execution paths of the program code, including paths that are rarely or never executed, along with all combinations of test case conditions.
Test Case Design	• Test cases are created based on the structure of the program code. • Common techniques used include path testing, condition testing, loop testing, and control flow testing.
Identification of Program Errors	• Detecting logical, syntax, and runtime errors in the program code. • Finding errors that may not be visible in black box testing.
Knowledge of Program Code	• Testers need to be familiar with the program code, architecture, and design of the software being tested. • Testing is restricted to individuals who understand code development.
Effectiveness of Testing	This is effective for ensuring that each part of the program code has been thoroughly tested.

Source: (Research Results, 2024)

RQ-2: What are the advantages of black box and white box approaches?

The advantages of black box and white box testing are summarized in Table 10.

Table 10. The advantages of black box and white box testing

Black Box Testing	White Box Testing
• Testing execution does not require programming expertise. Testers do not need to have knowledge of the code, enabling individuals with varying skill levels to perform the tests. • Focus on functionality, emphasizing the alignment of the software's features with user needs and established specifications. • The testing process is faster because an in-depth analysis of the code structure is not required. • Testing documentation is clear and detailed because testing is based on software specifications. This facilitates tracking and repeating tests in the future.	• Detecting hidden errors within the program's code structure. • Optimization of the code can be achieved by identifying inefficient parts of the code. • Testing can be conducted on boundary conditions and logical flows. • Testing can ensure that all conditions and edge cases are covered. • Coverage testing can be achieved by directly testing every statement and branch of the code.

Source: (Research Results, 2024)

RQ-3: What are the disadvantages of black box and white box approaches?

The disadvantages of black box and white box testing are summarized in Table 11.

Table 11. The disadvantages of black box and white box testing

Black Box Testing	White Box Testing
• Testing is limited to the surface of the software. This method may be ineffective in detecting errors hidden within the program's structure. • Limited test coverage occurs when testing fails to encompass all potential execution paths and boundary conditions within the software. • The success of testing heavily relies on the clarity and completeness of the software specifications. If the software specification documentation is incomplete, the implementation of testing will impact the software quality. • Difficulties arise in determining the appropriate test cases due to the challenge of identifying all possible inputs and suitable test scenarios, especially for complex applications with	• Testers must possess technical expertise in programming languages and system architecture. Testing execution may be impeded if resources are limited. • The testing process can be time-consuming, costly, and resource-intensive due to the complexity and detail required. • Dependency on the code exists. If the code undergoes changes, previously conducted tests may become invalid, necessitating updates to the testing. • The focus of testing is often too detailed on the internal structure of the program, potentially overlooking the software's functionality from the end user's perspective.

Black Box Testing	White Box Testing
many different input conditions.	

Source: (Research Results, 2024)

The implementation of black box testing and white box testing in software testing serves distinct purposes, particularly when the software being evaluated exhibits varying levels of complexity. A comparison of the implementation of black box and white box testing is described below.

1. Simple Complexity
   - Black box: the scope of testing focuses on ensuring that the main features conform to the specifications.
   - White box: Optimize the performance of the program code in critical sections, as the straightforward code structure aids testers in tracing each branch during test cases.
2. Intermediate Complexity
   - Black box: testing encompasses integration testing among functions for data input handling, complex usage scenarios, and overall system performance.
   - White box: Optimize the performance of sections of the program code that are executed frequently.
3. Complex Complexity
   - Black box: focus is on testing the overall functionality of the system, including rare scenarios and error conditions.
   - White box: resolve complex performance issues as the intricacy of the code structure increases. It is crucial to ensure that every path, branch, and loop has been thoroughly tested, as this process can become quite complex and time-consuming.

Black box testing is more commonly employed to evaluate functionality from the user's perspective. In contrast, white box testing offers deeper coverage by verifying the correctness of the logic and structure of the program code, which is especially crucial for software with high complexity. Ideally, both testing methods should be conducted concurrently to guarantee comprehensive software quality. This approach allows developers to ensure that the software not only operates correctly from an external viewpoint but also possesses a robust internal structure that is free from errors.

D. Data Synthesis

Based on the analysis data above, a comparison of the characteristics of black box and white box testing approaches in the context of software development projects can be concluded, as shown in Table 11. Due to the different objectives and characteristics of the two methods, black box testing and white box testing can be performed simultaneously to enhance the quality and reliability of the software by ensuring that all functional and structural aspects of the program have been thoroughly tested.

Examples of black box and white box testing have been conducted by several researchers, including Anisya Caty Praniffa et al. in 2023, who tested a parking system. The application of black box testing by Anisya aimed to evaluate the functionality of each feature, including the Login Page, Logout Page, Vehicle Entry Page, Active Vehicle Data Page, Parking Page, and Report Page (Praniffa et al., 2023). In contrast, white box testing was employed to assess the code structure within the main flow of the parking system application using the basis path testing technique (Praniffa et al., 2023). An example of a black box testing scenario is illustrated in Appendix A, while white box testing is depicted in Appendix B. The results of the study indicated that utilizing both methods enhances the objectives of test cases concerning system functionality and internal code structure.

Table 12. Differences between Black Box Testing and White Box Testing

Criteria	Black Box Testing	White Box Testing
Definition	A method of software testing that evaluates the functionality of the software by analyzing its inputs and outputs.	A method of software testing that analyzes the internal structure of the code.
Objective	To ensure that the software functions correctly and meets the specifications.	To verify that the logic of the program code functions correctly.
Testing Focus	Software functionality	Software design structure
Basis for Test Scenarios	Test scenarios are developed based on the software's functional requirements or specifications.	Test scenarios are developed based on the detailed design of the program code.
Application of Testing	Generally utilized in higher-level testing, such as system testing and acceptance testing.	Generally utilized in lower-level testing, such as unit testing and integration testing.

Criteria	Black Box Testing	White Box Testing
Example Techniques	Equivalence partitioning, boundary value analysis, decision table.	Statement coverage, branch coverage, basis path testing.
Understanding Programming Languages	Testers do not need to have an understanding of programming languages.	Testers need to have a deep understanding of programming languages.
Code Access	Testers do not have access to the program code.	Testers are granted access to the program code.
Tester Role	Independent tester.	Test engineer and programmer.
Advantages	Functional errors are easier to identify and comprehend from the end user's perspective.	Logical errors are identified, which enhances code quality and ensures thorough testing.
Disadvantages	The current testing approach does not cover all code paths, which makes it challenging to identify logical errors.	The task requires a high level of technical knowledge and can be more complex and time-consuming.
Applied to Various Types of Software Development Projects	• Commercial Software Development. • Projects with Time and Resource Constraints. • Applications with Complex User Interfaces (UIs).	• Critical Software Development with a Focus on High Security, Reliability, and Performance • Projects with Complex Code • Projects with Iterative Development Cycles

Source: (Research Results, 2024)

Another study by Sie in 2022 examined white box testing using the basis path testing technique. This approach involves creating scenarios based on the internal code structure (Appendix C) and utilizing a control flow graph, as illustrated in Appendix D. Test cases were subsequently developed using the paths identified in Appendix 4. During the execution of these test cases, certain paths could not be traversed due to specific conditions, particularly paths R4 and R7. This suggests that the program code is ineffective. Furthermore, some paths exhibited failure conditions. The final step involved correcting the program code based on the identified path anomalies and the paths that returned a failure status. The research findings indicate that the white box testing method is effective in identifying errors stemming from logical mistakes and internal code structures that may not be detected through black box testing.

CONCLUSION

In the world of software testing, white box testing and black box testing are two complementary methods. White box testing provides an in-depth analysis of the internal structure, while black box testing ensures functionality from the user's perspective. By understanding the strengths and weaknesses of each testing approach, development and testing teams can choose the method that best fits the needs and context of the software project they are working on. This literature review explores various studies and research discussing the objectives of testing, characteristics, differences, advantages, and disadvantages of these two testing methods.

Based on the literature review of both approaches, several practical recommendations for implementing black box testing methods for software developers include: (1) creating scenarios in black box testing that accurately reflect the actual usage patterns of end users when interacting with the software; (2) conducting integration testing between functions to ensure data accuracy; and (3) verifying that all software functionalities and error handling for user mistakes have been thoroughly tested. Meanwhile, practical recommendations for implementing white box testing methods include: (1) performing tests early at the smallest module (unit code); (2) ensuring that every path is executed with the assistance of code coverage tools; and (3) optimizing the program code whenever there is an unexecuted path.

The combination of both testing methods in the software development process can help developers ensure that the software functions correctly from the end user's perspective while maintaining high code quality. This combination allows for more comprehensive testing and error detection from both functional and structural aspects of the program.

REFERENCE

Amin, Z. M., Anwar, N., Shoid, M. S. M., & Samuri, S. (2022). Method for Conducting Systematic Literature Review (SLR) for Cyber Risk Assessment. *5th International Conference on Information Science*, *7*(SI10), 255–260. https://doi.org/10.21834/ebpj.v7isi10.4130

Amin, Z. M., Anwar, N., Shoid, M. S. M., & Samuri, S. (2023). A Systematic Literature Review for Modeling a Cyber Risk Assessment Framework. *Nd International Conference on Information Science, Technology, Management, Humanities, and Business*, *9*(SI18), 189–195. https://doi.org/10.21834/e-bpj.v9isi18.5481

Arifandi, A., Simamora, R. N. Z., Janitra, G. A., Yaqin, M. A., & Huda, M. M. (2022). Survei Teknik-Teknik Pengujian Software Menggunakan Metode Systematic Literature Review. *ILKOMNIKA: Journal of Computer Science and Applied Informatics E*, *4*(3), 297–315. https://doi.org/10.28926/ilkomnika.v4i3.436

Dhaifullah, I. R., Muttanifudin, M., Salsabila, A. A., & Yakin, M. A. (2022). Survei Teknik Pengujian Software. *JACIS : Journal Automation Computer Information System*, *2*(1), 31–38. https://doi.org/10.47134/jacis.v2i1.42

Faiza. Intan Mayla, Gunawan, & Andriani, W. (2022). Tinjauan Pustaka Sistematis: Penerapan Metode Machine Learning untuk Deteksi Bencana Banjir. *Jurnal Minfo Polgan*, *11*(2), 59–63. https://doi.org/10.33299/jpkop.22.2.1752

Fitriani, F., & Kuswadi, D. (2021). Driving factors of sustainable production in upland watersheds: systematic literature review. *E3S Web of Conferences*, *306*. https://doi.org/10.1051/e3sconf/202130604015

Golian, N. V, Golian, V., & Afanasieva, I. V. (2022). Black And White-Box Unit Testing For Web Applications. *Bulletin of National Technical University "KhPI". Series: System Analysis, Control and Information Technologies*, *7*(1), 79–83. https://doi.org/10.20998/2079-0023.2022.01.13

Hamzah, R. I. F., & Voutama, A. (2023). Pengujian Black Box Pada Aplikasi Database Perguruan Tinggi dengan Teknik Equivalence Partitions. *Antivirus : Jurnal Ilmiah Teknik Informatika*, *17*(1), 1–18. https://doi.org/10.35457/antivirus.v17i1.2501

Hartono, F. D., & Sugiarti, Y. (2022). Perbandingan Metode Equivalence Partitions Dan Boundary Value Analysis Pada Pengujian Black Box (Literatur Review). *Majalah Ilmiah METHODA*, *12*(2), 153–159. https://doi.org/https://doi.org/10.46880/methoda.Vol12No2.pp153-159

Iskandar, R. S. F., & Juandi, D. (2022). Study Literature Review: Realistic Mathematics Education Learning on Students' Mathematical Creative Thinking Ability. *SJME (Supremum Journal of Mathematics Education)*, *6*(1), 35–42. https://doi.org/10.35706/sjmev6i1.5739

Kalfin, Ibrahim, R. A., & Laksito, G. S. (2024). Optimization of White Box Testing by Utilizing Branching and Repeating Structures in Java Programs Using Base Path. *International Journal of Mathematics, Statistics, and Computing*, *2*(2), 85–89. https://doi.org/https://doi.org/10.46336/ijmsc.v2i2.98

Kurniawan, F., Khrisnawati, E. A., Hadiwiyanti, R., & Fitri, A. S. (2022). Pengujian Sistem Informasi Manajemen Siswa Berbasis Website Menggunakan Metode Black Box dan White Box. *Prosiding 2022 "Peran Teknologi dan Sistem Informasi Sebagai Daya Bangkit UKM Pasca Pandemi di Masa Adaptasi Kebiasaan baru,"* 246–261. https://doi.org/https://doi.org/10.33005/sitasi.v2i1.306

Kusum, Talwar, P., & Sharmila, K. (2024). Overview of Software Testing. *Global Journal of Engineering and Technology Advances*, *19*(1), 104–112. https://doi.org/10.30574/gjeta.2024.19.1.0060

Maulana, T. D., Nugroho, A. A. S., Suryaputra, B. A., & Wulansari, A. (2024). Tinjauan Literatur Sistematis: Manajemen Sumber Daya Ti Di Lingkungan Pendidikan. *Jurnal Mahasiswa Teknik Informatika*, *8*(1), 57–62. https://doi.org/10.36040/jati.v8i1.8717

Mengist, W., Soromessa, T., & Legese, G. (2020). Method for conducting systematic literature review and meta-analysis for environmental science research. *MethodsX*, *7*(100777). https://www.sciencedirect.com/science/article/pii/S221501611930353X

Mishra, D. B., Acharya, A. A., & Acharya, S. (2020). White Box Testing Using Genetic Algorithm—An Extensive Study. In J. Singh, S. Bilgaiyan, B. S. P. Mishra, & S. Dehuri (Eds.), *A Journey Towards Bio-inspired Techniques in Software Engineering* (Vol. 185, pp. 167–188). Intelligent Systems Reference Library. https://doi.org/https://doi.org/10.1007/978-3-030-40928-9_9

Praniffa, A. C., Syahri, A., Sandes, F., Fariha, U., Giansyah, Q. A., & Hamzah, M. L. (2023). Pengujian Black Box Dan White Box Sistem Informasi Parkir Berbasis Web Black Box And White Box Testing Of Web-Based Parking Information System. *Jurnal Testing dan Implementasi Sistem Informasi*, *1*(1), 1–16. https://www.journal.al-matani.com/index.php/jtisi/article/view/321

Reyna, R. S., Aleu, F. G., Gutierrez, E. M. A. G., Ramirez, J. D., Reyes, J. A. G., & Kumar, A. (2020). A systematic literature review of data science, data analytics and machine learning applied to healthcare engineering systems. *Management Decision*, *60*(2), 300–319. https://doi.org/https://doi.org/10.1108/MD-01-2020-0035

Riyadi, S. (2023). Adopsi Metode DevOps Sebagai Acuan Pengembangan Aplikasi Bantuan Hukum. *Generation Journal*, *7*(1), 23–70. https://doi.org/https://doi.org/10.29407/gj.v7i1.19629

Salim, S. W. G. A., Ibrahim, R., & Wahab, J. A. (2021). Comparative Analysis of Software Testing Techniques for Mobile Applications. *Journal of Physics: Conference Series*, *1793*(1), 1–11. https://doi.org/10.1088/1742-6596/1793/1/012036

Santi, P. A. D. A., Afwani, R., Albar, Moh. A., Anjarwani, S. E., & Mardiansyah, A. Z. (2022). Black Box Testing with Equivalence Partitioning and Boundary Value Analysis Methods (Study Case: Academic Information System of Mataram University). In *Proceedings of the First Mandalika International Multi-Conference on Science and Engineering 2022, MIMSE 2022 (Informatics and Computer Science)* (pp. 207–219). Atlantis Press International BV. https://doi.org/10.2991/978-94-6463-084-8_19

Sasmito, G. W. (2020). White Box Testing with Basis Path Technique in the Demography Administration Website. *International Conference on Sustainable Engineering and Creative Computing (ICSECC)*, 86–92. https://doi.org/https://doi.org/10.1109/ICSECC51444.2020.9557428

Shadiq, J., Safei, A., & Loly, R. W. R. (2021). Pengujian Aplikasi Peminjaman Kendaraan Operasional Kantor Menggunakan BlackBox Testing. *Information Management for Educators and Professionals*, *5*(2), 97–110. https://doi.org/https://doi.org/10.51211/imbi.v5i2.1561

Sholeh, M., Gisfas, I., Cahiman, & Fauzi, M. A. (2021). Black Box Testing on ukmbantul.com Page with Boundary Value Analysis and Equivalence Partitioning Methods. *Journal of Physics: Conference Series*, *1823*(1), 1–8. https://doi.org/10.1088/1742-6596/1823/1/012029

Sie, J. B. L., Musdar, I. A., & Bahri, S. (2022). Pengujian White Box Testing Terhadap Website Room Menggunakan Teknik Basis Path. *Jurnal KHARISMA Tech*, *17*(2), 45–57. https://doi.org/10.55645/kharismatech.v17i2.235

Suryani, I., Misrita, M., & Ristati, R. (2021). Folklore and It's Effect on Student's Ability In Reading Narrative Text: a Systematic Literature Review. *Indonesian Language Education and Literature*, *7*(1), 194–205. https://doi.org/10.24235/ileal.v7i1.9089

Susilawati, F. E., Suparman, & Patalo, A. M. G. (2022). Pengujian Black Box Aplikasi Penjualan Pupuk Bersubsidi Menggunakan Teknik Equivalence Partitioning. *Jurnal Ilmiah Information Technology d'Computare*, *12*(2), 30–34. https://doi.org/10.30605/dcomputare.v12i2.48

Taley, D. S., & Pathak, B. (2020). Comprehensive Study of Software Testing Techniques and Strategies: A Review. *International Journal of Engineering Research & Technology (IJERT)*, *9*(8), 817–822. https://doi.org/10.17577/IJERTV9IS080373

Tanli, M., Jiang, Y., Wang, X., & Peng, R. (2020). Black-box approach for software testing based on fat-property. *International Conference on Computer Science Communication and Network Security*. https://doi.org/https://doi.org/10.1051/matecconf/202030902008

Uminingsih, Ichsanudin, M. N., Yusuf, M., & Suraya. (2022). Pengujian Fungsional Perangkat Lunak Sistem Informasi Perpustakaan Dengan Metode Black Box Testing Bagi Pemula. *Jurnal Ilmiah Teknik Dan Ilmu Komputer*, *1*(2), 1–8. https://doi.org/10.55123/storage.v1i2.270

Wazzan, M., Algazzawi, D., Bamasaq, O., Albeshri, A., & Cheng, L. (2021). Internet of things botnet detection approaches: Analysis and recommendations for future research. *Applied Sciences (Switzerland)*, *11*(12), 1–46. https://doi.org/10.3390/app11125713

Weber, M., Apel, S., & Siegmund, N. (2021). White-box performance-influence models: A profiling and learning approach. *Proceedings - International Conference on Software Engineering*, 1059–1071. https://doi.org/10.1109/ICSE43902.2021.00099

Wijaya, Y. D., & Astuti, M. W. (2021). Pengujian Blackbox Sistem Informasi Penilaian Kinerja Karyawan PT Inka (Persero) Berbasis Equivalence Partitions. *Jurnal Digital Teknologi Informasi*, *4*(1), 22–26. https://doi.org/10.32502/digital.v4i1.3163

Wintana, D., Pribadi, D., & Nurhadi, M. Y. (2022). Analisis Perbandingan Efektifitas White-Box Testing dan Black-Box Testing. *Jurnal Ladang Artikel Ilmu Komputer*, *2*(1), 8–16. https://doi.org/10.31294/larik.v2i1.1382